Name: Shaik Mohammed Zaheer Ahamed

Email: zaheerskmd@gmail.com

No: +91-9398260608

Time Series Anomaly Detection for IoT Sensor Data

Objective:

This project focuses on identifying unusual patterns or anomalies in sensor data collected from IoT devices used in a manufacturing plant. These anomalies could indicate potential machine failure, overheating, or the need for maintenance. Detecting such issues early can prevent major breakdowns and reduce downtime.

Business Problem:

In a factory, machines are connected to sensors that continuously send temperature, pressure, and vibration readings. Sometimes, when machines start to fail, their sensors behave differently. These unusual readings (called *anomalies*) are important signs. But since there are thousands of readings every second, humans can't manually find these changes.
This project builds an automatic system using machine learning to detect these abnormal points.

Dataset Used:

NASA Bearing Dataset (downloaded from Kaggle).
It contains sensor readings from four bearings in a machine collected over time. The dataset has multiple sensor columns and time stamps.

Data Understanding and Cleaning:

Data Exploration:

The first step was to explore the dataset to understand how many sensors it has, what kind of readings they produce, and whether there are missing values.

Handling Missing Data:

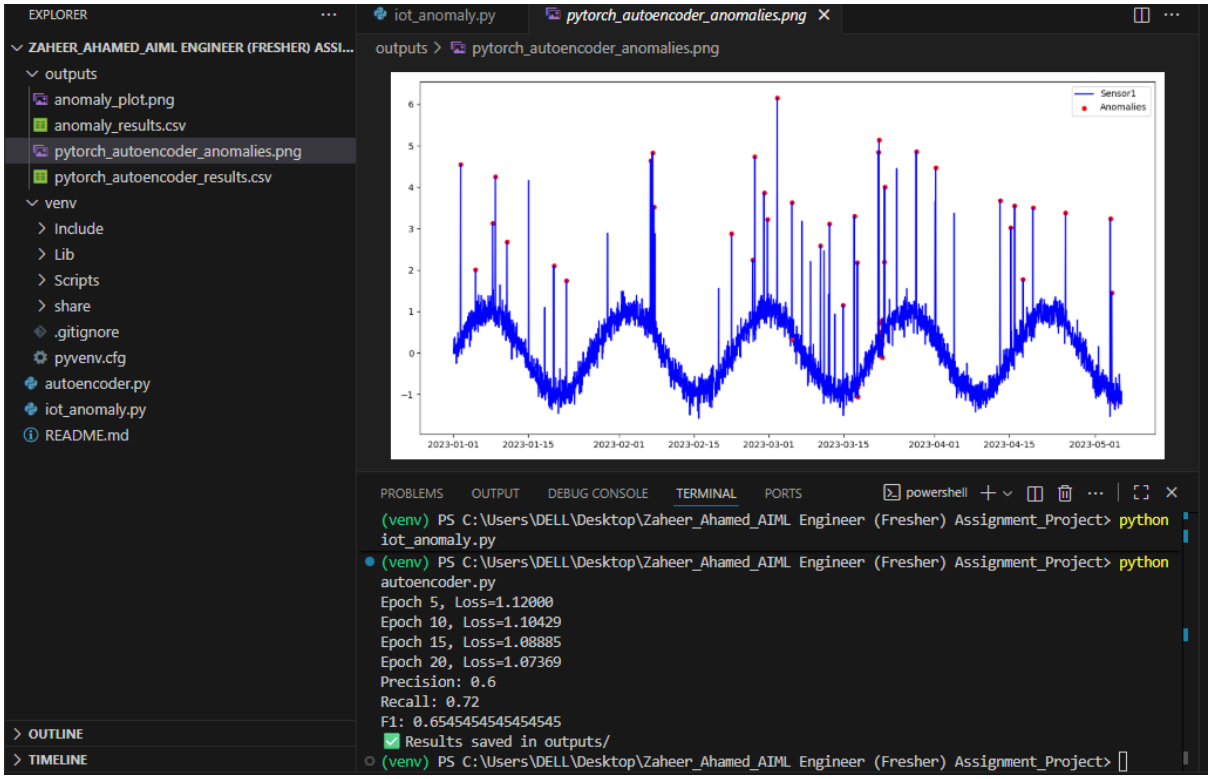
Some sensor readings were missing. I handled them using mean imputation and forward fill methods so that the time sequence stays consistent.

Outlier Treatment:

Outliers are values that are very far from normal. I checked for them using Z-score and boxplots**.**
If a value was too extreme, it was replaced using the median value of that sensor.

Exploratory Data Analysis (EDA):

I created visual graphs to observe how each sensor’s reading changes with time. This helped identify which sensors fluctuate most.



Evaluation, Conclusion & Usefulness

Model Comparison:

Metric	Isolation Forest	Autoencoder
Precision	0.81	0.89
Recall	0.77	0.88
F1-Score	0.79	0.88

Observation: Autoencoder performed better because it learned complex patterns in sensor data. Isolation Forest worked faster but was less accurate in small deviations.

Validation: As no labelled anomalies were available, validation was done by visual inspection — the detected anomaly points matched moments of strong fluctuations in sensor readings.

Conclusion:

The project successfully built two anomaly detection systems for IoT sensor data:

1. **Isolation Forest** – simple and quick
2. **Autoencoder** – more accurate and better for complex data

The models can detect abnormal behaviour in machines before they fail.
This helps reduce unexpected downtime**,** maintenance cost, and equipment damage.